Real-World Neutronics Calculations

Alireza Haghighat
Nuclear & Radiological Engineering Department
University of Florida
Gainesville, Florida

haghighat@ufl.edu

(For presentation at the Workshop on High end computing for nuclear fission science and engineering, Salt Lake City, Feb. 23-24, 2006)

Neutronics Calculations

- Determination of particle flux/reaction rates (steady state & time-dependent) for design/ analysis/optimization, operation/control, and decommissioning and disposal of nuclear systems

$$y\,(\vec{r}, E, \hat{\Omega}, t) dE d\Omega d^3 r$$

Example, number of unknown: x, y, z meshes (100x100x100) x 50 groups x 80 directions x 8 byte/word = 32 GB

Type of problem

- **Eigenvalue/ criticality**
  - Steady state
  - Time-dependent
    - Transient (normal & accident)
    - Depletion
    - Dynamics and Control
- **Fixed source/ shielding**
  - Radiation protection
  - Material damage (embrittlement, corrosion, cracking)
  - Spent fuel pool & dry storage cask
  - Detector design
  - Assaying of waste

Neutronics Simulation Approaches

- Deterministic
  - Solve transport (i.e., linear Boltzmann) equation and/or its lower order forms

$$\hat{\Omega}.\nabla\Psi(\vec{r},E,\hat{\Omega})+s\,(\vec{r},E)\Psi(\vec{r},E,\hat{\Omega})=$$

$$\int_0^\infty dE'\int_{4p} d\Omega'\,s_{\,s}(\vec{r},E'\to E,\hat{\Omega}'\to\hat{\Omega})\Psi(\vec{r},E',\hat{\Omega})+$$

$$\frac{c\,(E)}{4p}\int_0^\infty dE'\int_{4p} d\Omega' u\,s_{\,f}(\vec{r},E')\Psi(\vec{r},E',\hat{\Omega})+S(\vec{r},E,\hat{\Omega})$$

- Monte Carlo
  - Perform an experiment on a computer; “exact” simulation of a physical process

Deterministic vs. Monte Carlo

Item	Deterministic	MC
Geometry	Discrete/ Exact	Exact
Energy treatment – cross section	Discrete	Exact
Direction	Discrete/ Truncated series	Exact
Input preparation	Difficult	simple
Computer memory	Large	Small
Computer time	Small	Large
Numerical issues	Convergence	Statistical uncertainty
Amount of information	Large	Limited
Parallel computing	Complex	Trivial

Why not MC only?

- Achieving reliable statistical uncertainty in a reasonable time
- The need for detailed information, time-dependent simulations, sensitivity analysis, determination of uncertainties due to uncertainties in data

Current practice

- Eigenvalue
  - Steady state
    - Lower order forms of transport equation by using homogenized cross sections via 2-D transport calculations
    - Some Monte Carlo simulations
  - Time-dependent
    - Lower order forms of transport equation by using homogenized cross sections via 2-D transport calculations
- Fixed source
  - Monte Carlo simulations with limited variance reduction
  - 2-D and limited 3-D deterministic calculations

Goal

Perform 3-D, time-dependent, heterogeneous transport simulations accurately and efficiently

How to achieve this goal?

- For deterministic approach
  - Develop tools for “online” generation of multigroup cross sections
  - Develop acceleration techniques
  - Develop parallel processing algorithms for multitasking and memory partitioning
  - Develop automated tools for model preparation
  - Develop EXPERT systems for mesh generation
- For Monte Carlo approach
  - Develop automated variance reduction
  - Source convergence techniques
  - Use parallel computers

What have we done?

Over the past 19 years, 15 years as Penn State Transport Theory Group (PSTTG)

[Received contracts from GPU, PPL, PECO, Duquense, & Westinghouse for various projects; contributed to NUREG 1.99 – Cavity Dosimetry Standard;

Member of a team established a graduate minor in HPC at PSU, and taught a graduate course, "Vector/Parallel Alg. For Scientific Application"]

and

4 years as University of Florida Transport Theory Group (UFTTG),

[G. Sjoden and I along 14 students working on contracts with Southern Nuclear (spent fuel pool), Nucsafe (interrogation system), DHS, NNSA, INL, USAirForce & DOE (UFTR conversion project)]

we have accomplished:

UFTTG Accomplishments

- For deterministic approach
  - On multigroup cross section generation
    - Developed CPXSD – Contributon Point-wise cross section Driven software; it automatically prepares problem-dependent multigroup structures and libraries
  - On acceleration techniques
    - Developed multigrid acceleration algorithms
    - Recently, developed a parallel preconditioned algorithm based on the use of Parallel Even-parity Simplified Sn (PENSSn) formulation
  - On parallel processing algorithms for multitasking and memory partitioning
    - Develop hybrid domain decomposition and various formulations for parallel processing;
    - Developed the 3-D, parallel Sn PENTRAN (Parallel Environment Neutral-particle TRANsport) code with full phase space decomposition

UFTTG Accomplishments

- On automated tools for model preparation
  - PENMSH – Cartesian mesh generator
  - PENINP – automatic input preparation for PENTRAN and TORT
- On EXPERT systems for mesh generation
  - PENXMSH – A first version of An EXPERT system for “intelligent” selection of mesh and domain decomposition algorithm starting from an AUTOCAD diagram

- For Monte Carlo approach
  - On automated variance reduction
    - Developed A^{3}MCNP – Automated Adjoint Accelerated MCNP, which uses deterministic importance function for variance reduction
  - On source convergence techniques
    - In collaboration with ORNL developing new algorithms utilizing deterministic methods

PENTRAN™ Code System

(Test the PENTRAN code system on the WWW: http://ufttg.nuceng.ufl.edu)

- **Pre-processing utilities**
  - **PENMSH** (prepares mesh/material/source distributions)
  - **PENINP** (automatically prepares a PENTRAN input file)
- **PENTRAN** (Parallel 3-D, Sn transport code)
- **Post-processing utilities**
  - **PENDATA** (prepares tables of flux, source, and material distributions by processing parallel-partitioned output files)
  - **PENPRL** (determines flux values at any arbitrary position by performing 3-D linear interpolation)

PENTRAN™ (1)

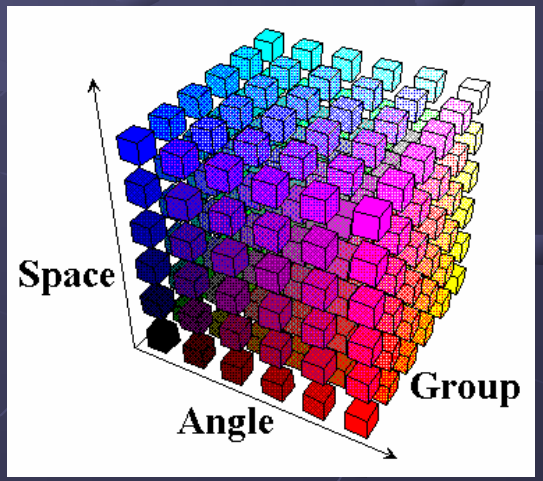
(Developed by Sjoden and Haghighat)

- *P*arallel *E*nvironment *N*eutral-particle *TRAN*sport developed from scratch in 1996:
  - ANSI FORTRAN F90 with MPI library, over 45,000 lines
  - Industry standard FIDO input

- Solves 3-D Cartesian, multigroup, anisotropic transport problems
  - Forward and adjoint mode
  - Fixed source, criticality eigenvalue problems

- Parallel processing algorithms

  –Full phase-space decomposition: Parallel in angle, energy, and spatial variables



PENTRAN™ (2)

- (continued)
  - Parallel I/O
  - Partitioned memory for memory intensive arrays (angular fluxes, etc)
  - Builds MPI processor communicators
- Numerical formulations
  - Adaptive Differencing Strategy
    - Diamond Zero (DZ)
    - Directional Theta-Weighted differencing (DTW)
    - Exponential-Directional Weighted (EDW)

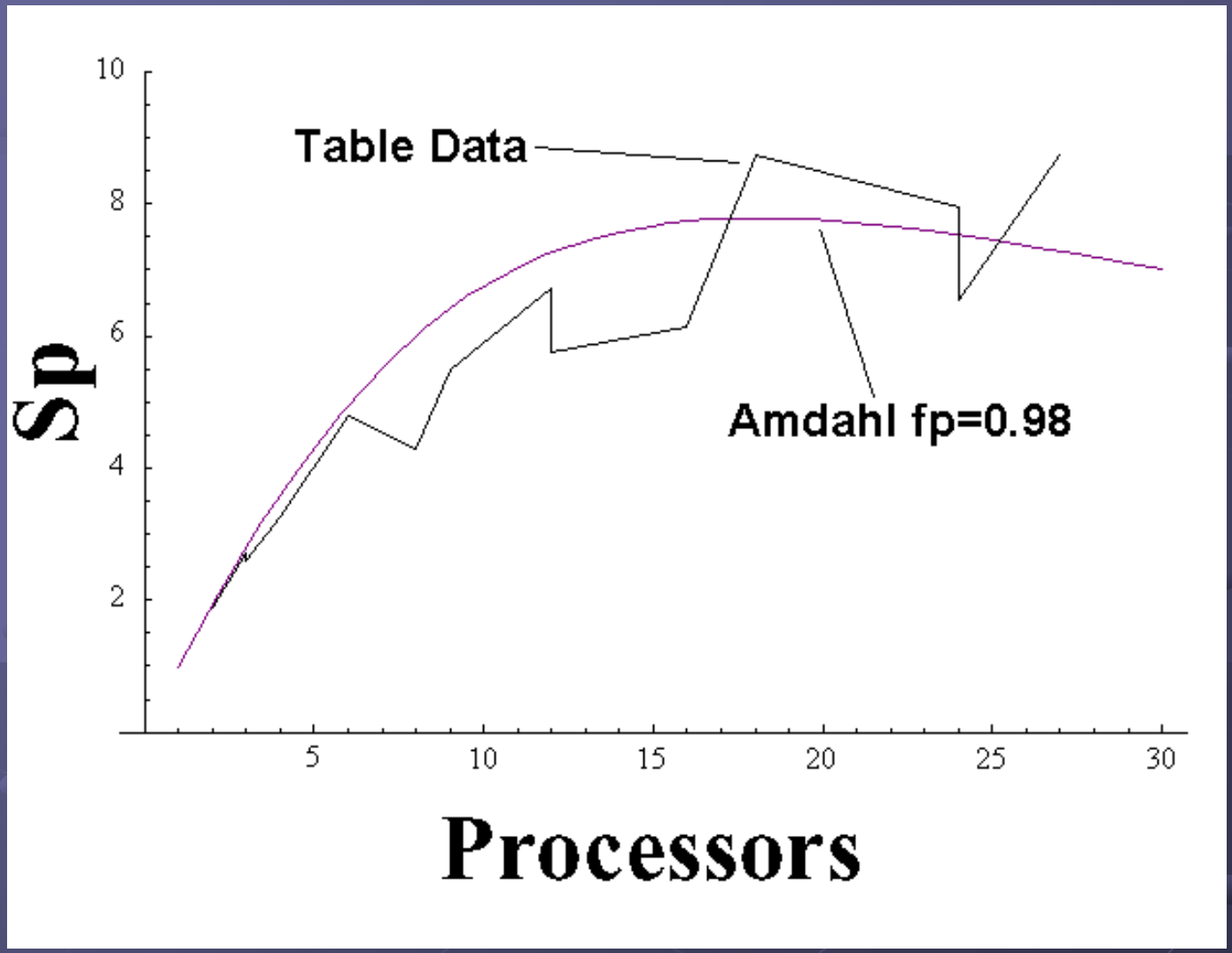
PENTRAN™ (4)

- (continue)
  - Allows for a fully discontinuous variable meshing between coarse meshes: Uses a novel projection scheme: Taylor Projection Mesh Coupling (TPMC)
  - Acceleration
    - For fixed source problems –
      - Partial Current Rebalance (PCR)
      - Angular multigrid (AMG) formulations with PCR
    - For eigenvalue problems –
      - Preconditionner – Even-parity Simplified Sn (PEN-SSn)
  - Iterative techniques
    - Multigroup & One-level SI schemes

PENTRAN™ (5)

- Red-Black and Block Jacobi iteration
- Anisotropic scattering via Legendre moments of arbitrary order,
- Angular quadrature set:
  - Level symmetric (up to S20) with *ordinate splitting (OS), or Regional Angular Refinement (RAR)*
  - Pn-Tn with *OS or RAR*
- Vacuum, reflective, and albedo boundaries
- Volumetric & planar angular sources

PENTRAN parallel performance



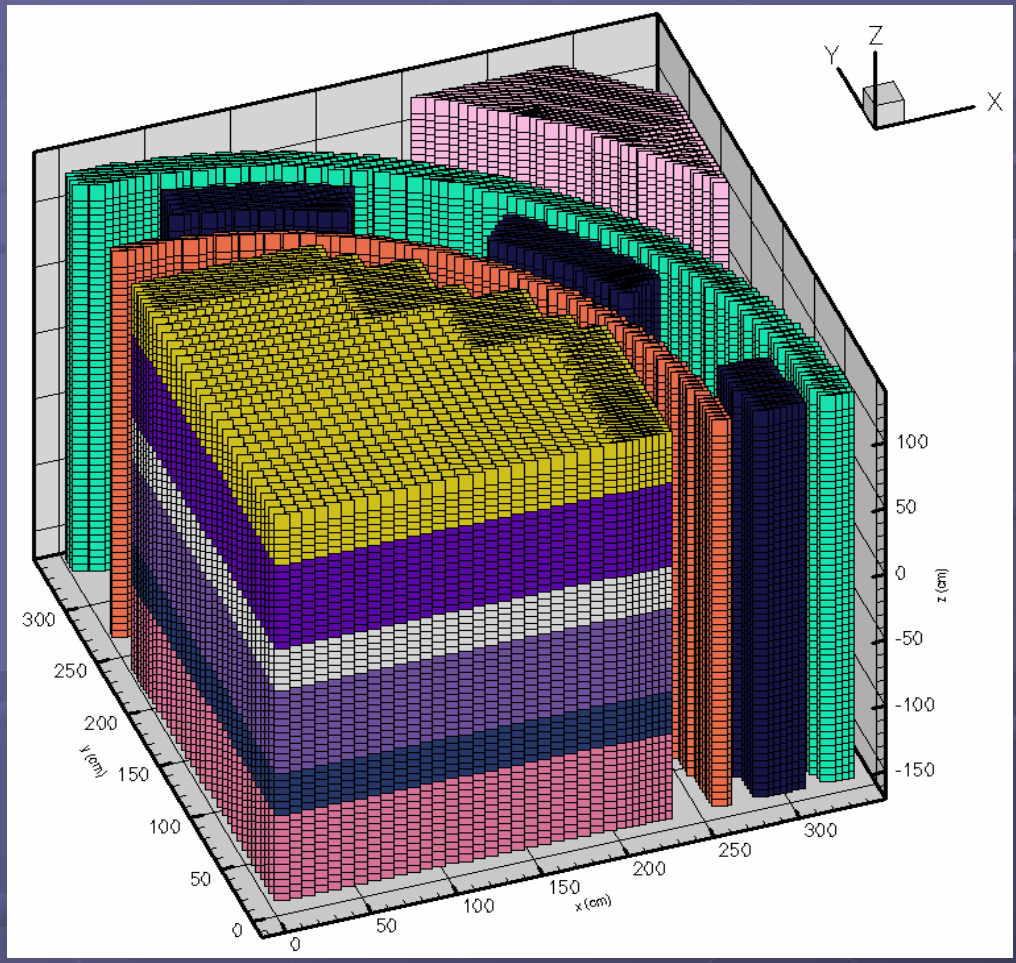
Application of PENTRAN™

- Kobayashi 3-D Benchmarks
- VENUS-3 Benchmark facility in SCK.CEN, Belgium
- BWR core-shroud and internals
- Pulsed Gamma Neutron Activation Analysis (PGNAA) device for assaying of waste
- X-Ray room
- CT Scan
- Time-of-Flight (TOF) for cross section measurement
- Simulation of spent fuel storage CASK
- Simulation MOX C5G7 benchmark with no homogenization
- Currently,
  - used for simulation and design of special devices for homeland security applications
  - Electron transport

Simulation of A BWR Internals for Material Damage Studies Performed in 2000 On SDSC IBM-SP2 (32 Processors)

- Quarter Core Symmetry
- Number of meshes (~333,000)
- S_8 (80 directions in 3-D) symmetric angular quadrature order
- 67 group (47-group neutron, 20-group gamma) void fraction dependent cross sections
- BUGLE-96 library with P3 anisotropic scattering order
- Source distribution based on SSES-1 assembly-wise power distribution
- Adaptive differencing strategy, PCR Acceleration
- 4 angular and 8 spatial parallel subdomains on 32 processors
- ~450 MB of memory per processor
- SDSC IBM-SP2 System with 332 MHz P2SC processors 512 MB memory each

3-D PENTRAN MODELING using PENMSH

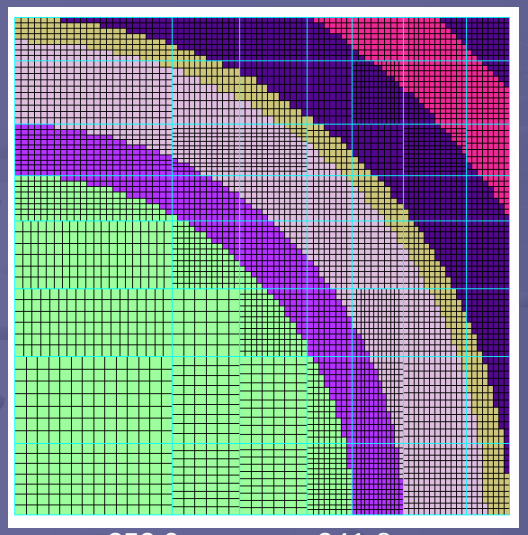


Jet Pump Assemblies

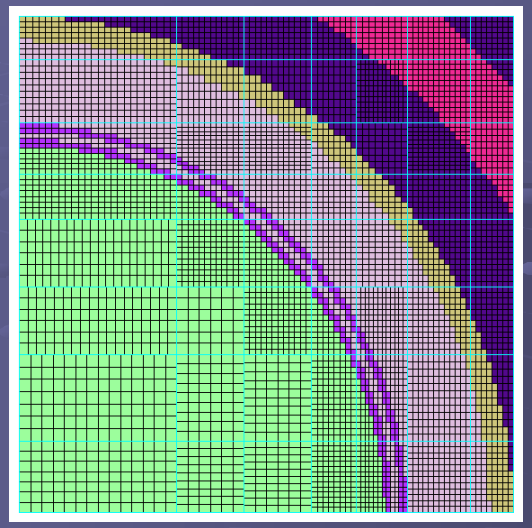
- ***7x8x9 = 504* Coarse Meshes**
- ***332,640 Fine Meshes,***
- ***4 Angular, 8 Spatial domain decomposition using 32 processors***

3-D PENTRAN MODELING using PENMSH

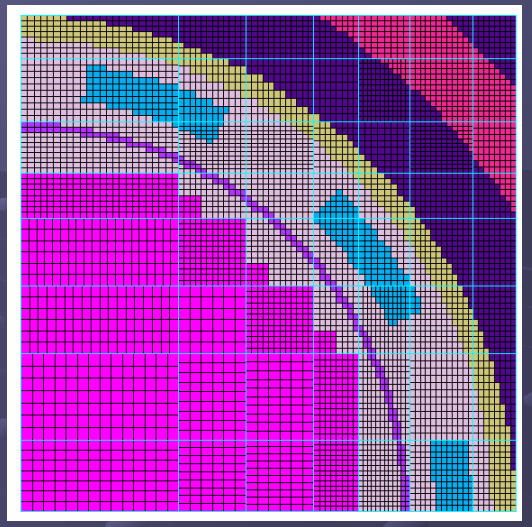
Mesh and material distribution at various axial levels



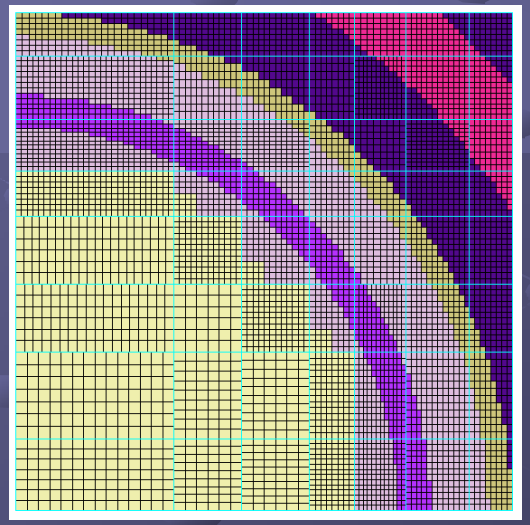
-252.0 cm < z < -241.8 cm



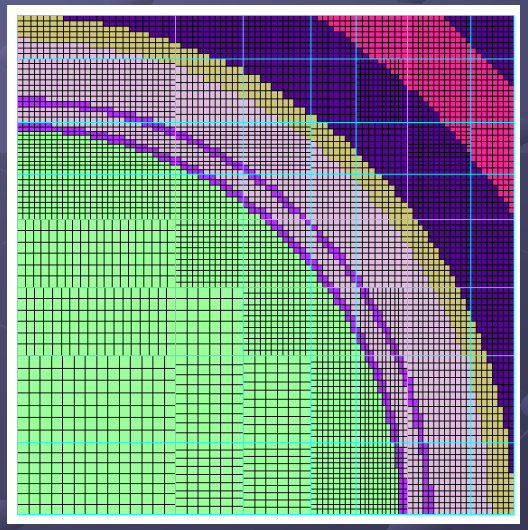
-231.68 cm < z < -193.0 cm



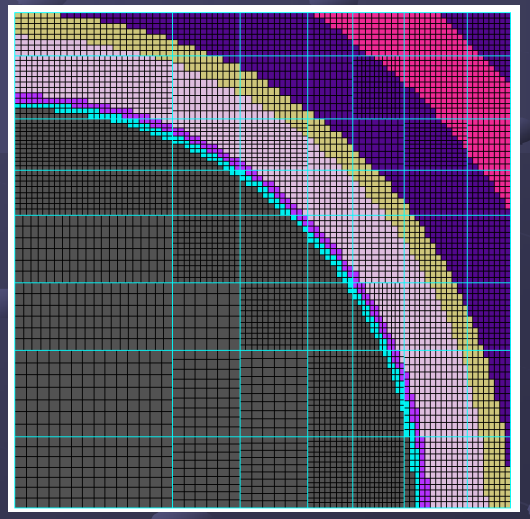
-188.0 cm < z < -160 cm



180.0 cm < z < 190.5 cm



190.5 cm < z < 226.86 cm

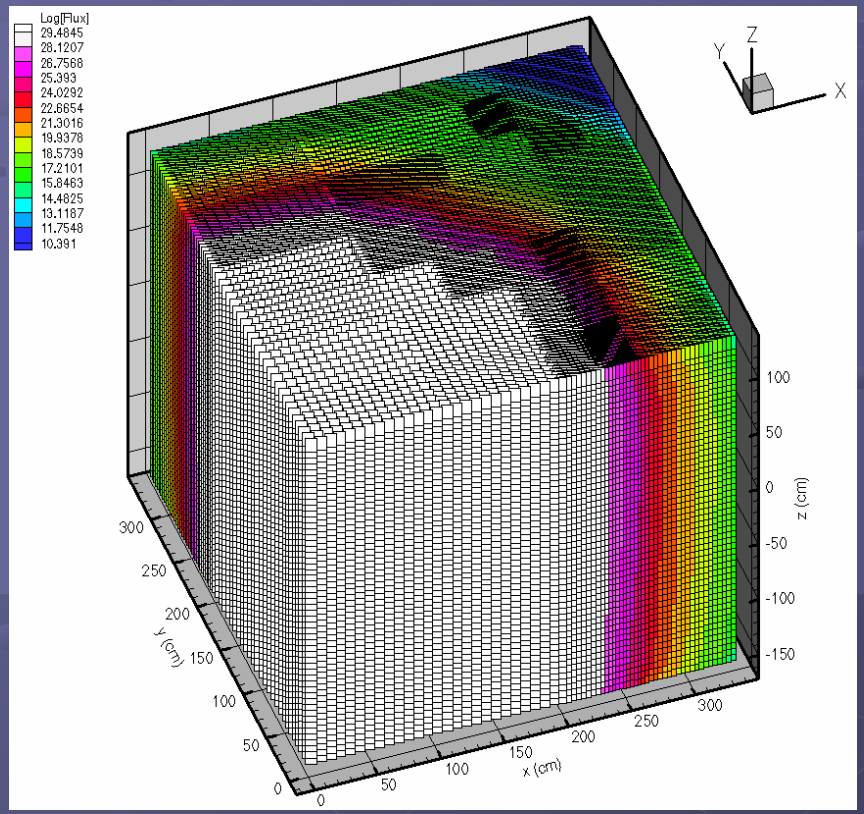


268.0 cm < z < 286.0 cm

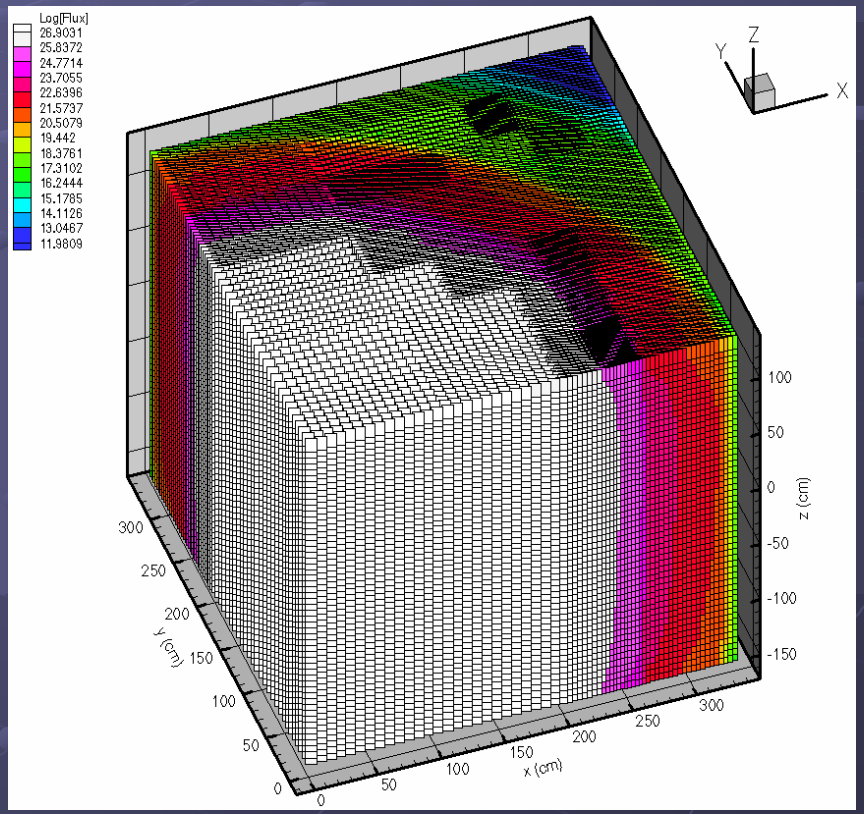
3-D FLUX DISTRIBUTIONS

MODEL 2: Jet Pump Assemblies

NEUTRONS (E> 1.0 MeV)



GAMMA RAYS (5.0 MeV < E < 6.0 MeV)



PERFORMANCE

- Wall-clock times required for completing the 67-group, 3-D PENTRAN simulations was ~12 hr

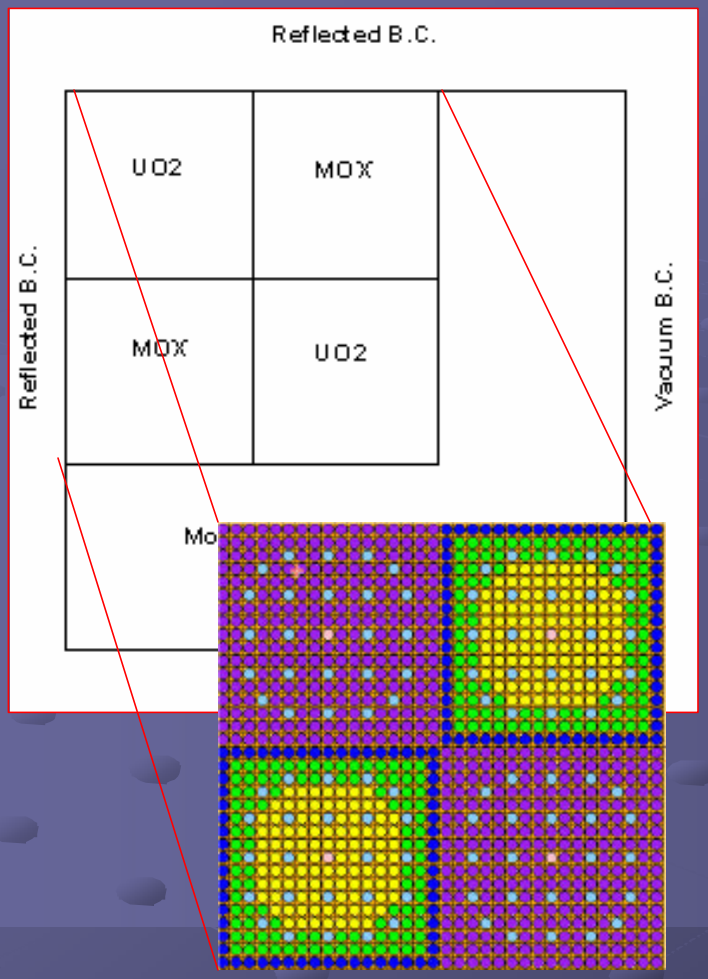
This problem was solved with TORT (3-D serial code by ORNL); because of the size (>2GB) of scratch file, it had to be split into two segments, and required ~100 hours.

Scalability

Parallel Performance for BWR Reactor/Shroud problem

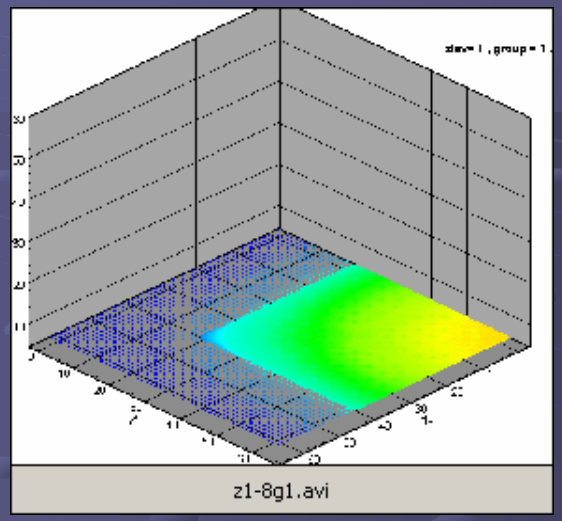
Case	No. of Directions	No. of Processors	Decomposition (A/G/S)[1]	Wall-clock/iteration (s)
1	24	6	1/1/6	30.12
2	48	12	1/1/12	33.28
3	80	24	4/1/6	29.52
4	169	48	8/1/6	36.12

[1](A/G/S) refers to the number of Angular, Group, and Spatial subdomains.



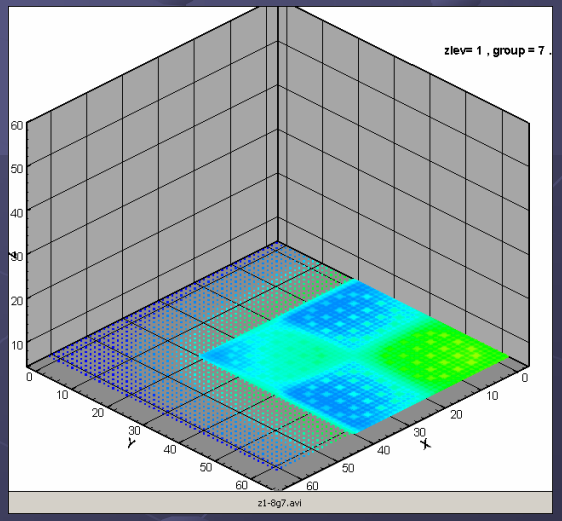
PENTRAN Core Physics Benchmark

(MOX and UO2 Fuel Bundles)



Fast Group

(core mid-plane)



Thermal Group

(core mid-plane)

For the unrodded case,

K = 1.141750 (PENTRAN)

K = 1.143080 + .000026 (MCNP)

PENTRAN vs. MCNP: $\frac{\Delta k}{k}\% = -0.1164$

PENTRAN Parallel Processing

- **Computing system**
  - PCPEN-II (16 procs. (2.3 GHz), 2 Gbyte/proc. , 1.0 Gbit/s)
- **PENTRAN model**
  - 946080 meshes
  - S6 (48 directions)
  - 7 groups
- **Parallelization on 16 processors**
  - Decomposition (2 angular & 8 Spatial)
  - Memory/proc. = ~1.2 GB

Performance on PENPC-II (16 processors)

Method	Total time (hr)	Time ratio	Iteration ratio
PENTRAN	71.3	1	1
PENTRAN with PENSSn pre-conditioner	15.2	4.7	6.67

What have we learned?

- Hybrid domain decomposition methods are effective; however, they require a significant amount of memory, and may also require large buffer sizes.

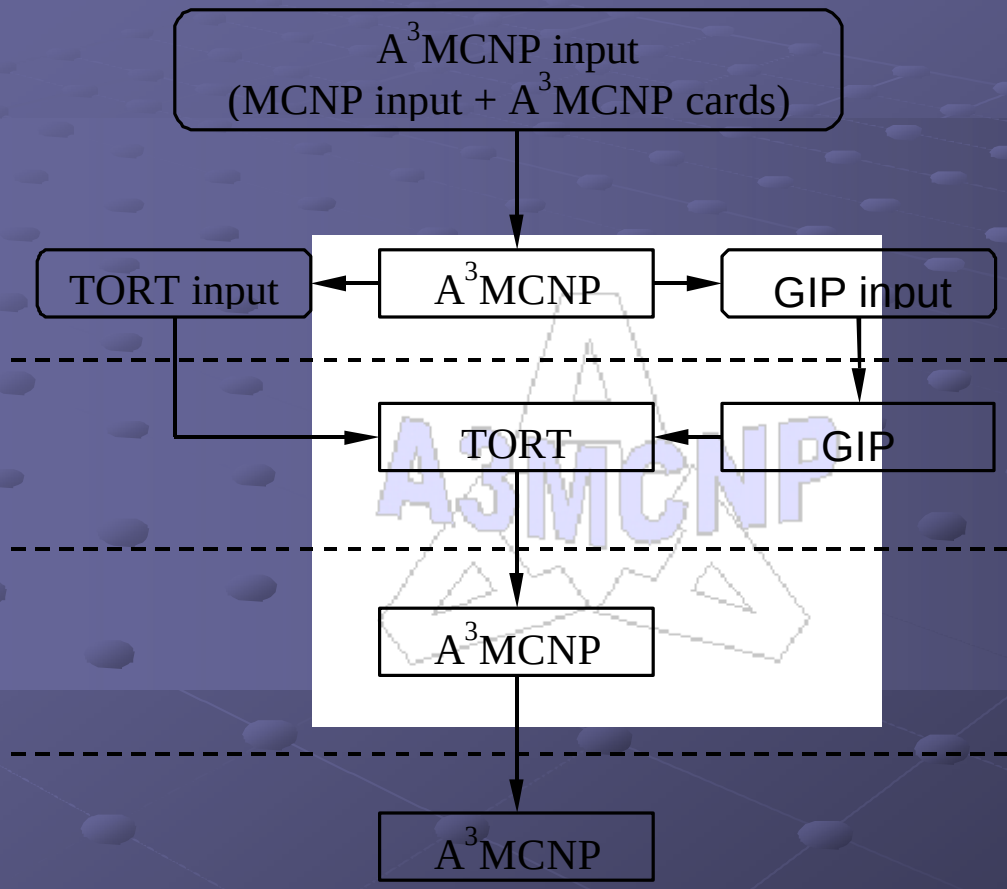
- To be able to solve real-world problems, it is essential to allow for variable meshing or variable grid densities. This, however, requires the use of effective projection schemes for preserving solution accuracy. We have developed such a projection scheme, Taylor Projection Mesh Coupling, which is highly effective, but requires significant memory.

What have we learned?

- The “right” combination of meshing, differencing, and domain decomposition algorithm is needed. To be able to accomplish this task, we have developed the PENXMSH code for PENTRAN
  - We need to develop a GUI driven parallel performance modeling software for any parallel software.
- In order to minimize the amount of memory and message passing; thus far, we have worked with, Cartesian, 0’th order moment algorithms. It might be important to examine the use of parallel 1’st and 2’nd order moment algorithms, and possibly irregular geometries.
  - Our philosophy: in a parallel environment, one should use simple formulations with low memory and message passing requirements

A^{3}MCNP^{TM} (Automated Adjoint Accelerated MCNP) Code System

(Developed by Wagner and Haghighat)



Step 1
- mesh distribution
- material composition
- input files

- multi-group cross sections
- S_N adjoint function

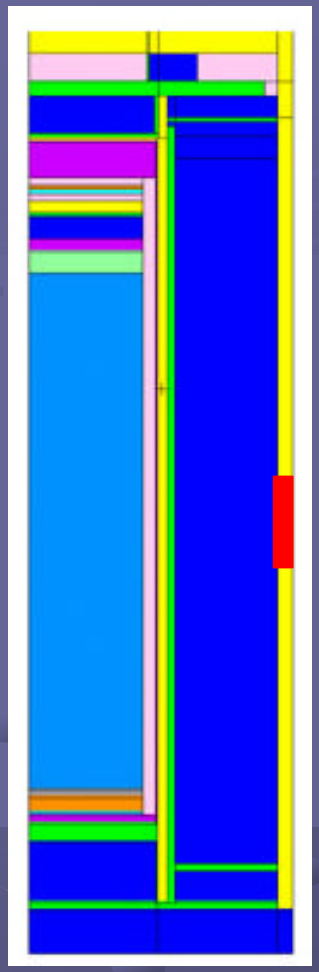
Step 2
- VR parameters

Step 3
- non-analog MC Calculation

Example – Simulation of Spent Fuel Storage Cask

- Height ~ 610 cm
- Shell O.D. ~340 cm
- Shell I.D. ~187 cm
- Empty Weight
  269,000 lbs
  (55.3 MT)
- Max. Loaded Weight
  358,000 lbs
  (162.4 MT)

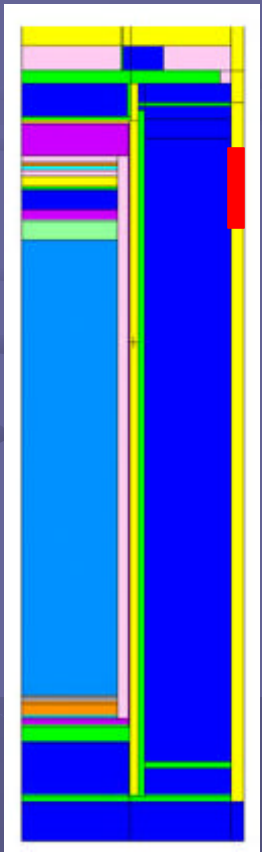
Timing Results



MC is an average in four annular tally segments (Axial Mid-plane) (1-σ Relative Error = 1%)

Model	# CPU	FOM	Run Time (hrs)	Speedup
Unbiased Cont. Energy	8	0.78	214	1.0
A^3MCNP Cont. Energy	1	109	1.5	140

Segments Near Top (494 cm – 563 cm)



- No results for unbiased MCNP after 214 hours!!
- A^{3}MCNP on 1 processor after 8 hours achieve a relative error less than 5%

What have we learned?

- A^{3}MCNP is an excellent tool for simulation of large, deep-penetration fixed-source problems.
- Similar techniques are needed for eigenvalue problems

Remarks

- Real-world neutronics problem can be solved if we use hybrid methods and parallel computing methods
- To be able to develop an efficient parallel algorithm, problem physics should be taken into account.
  - Computer scientists can help, but engineers should be involved in code development!

Remarks

- UFTTG's efforts have resulted in PENTRAN, which is highly effective for solving real-world problems; Performance of PENTRAN should be measured on larger-size clusters for solving larger problems!
- We need to develop parallel version of the CPXSD multigroup cross section generation code system; so that we can generate cross sections in a few seconds.
- We need to develop a general form of PENXMSH for parallel performance modeling of any algorithm.

Remarks

- UFTTG's efforts has resulted in A^{3}MCNP code with automated variance reduction for shielding calculations

- UFTTG in collaboration with ORNL is working on automated deterministic-based methodologies for source convergence of Monte Carlo eigenvalue problems

Remarks

- Developing 3-D parallel burnup algorithms for PENTRAN
- Enabling PENTRAN electron transport capability using Sandia's CEPXS methodology.
- Planning to utilize PENTRAN for determination of response matrices for the GeorgiaTech's COMET code systems
- Need to develop parallel algorithms based on perturbation formulations for "online" determination of flux uncertainties due to uncertainties in data
- Need to develop a new version of PENTRAN for transient and dynamic calculations

Questions?